Assignment 6: The Laplace Transform

Jnaneswara Rao Rompilli [EE20B052]

March 26, 2022

Abstract

- Forced oscillatory system
- Solve coupled spring problem
- Find Steady State Transfer Function of RLC circuit and find output response

1 Introduction

In this assignment, we will analyze Linear Time-Invariant systems using scipy library in python. Our analysis will be mostly based on polynomial transfer functions

2 Tasks

2.1 Forced oscillatory system

2.1.1 Impulse response

The system is characterized by given differential equation

$$\frac{\mathrm{d}^2x}{\mathrm{d}t^2} + 2.25x = f(t) \tag{1}$$

$f(t)$ is defined as,

$$f(t) = cos(\omega t) * \exp(-at)u(t) \tag{2}$$

The laplace transform of input signal is,

$$F(s) = \frac{s+a}{(s+a)^2 + 2.25} \tag{3}$$

In the above equation 'a' represents decay value. We will consider two different values 0.5, 0.05 for a and analyze the impulse response

```
# Plot impulse response for given transfer function with different decay values
def plot_impulse(decay, level, cnt):
    H = sp.lti(
        [1, decay], np.polymul([1, 0, 2.25], [1, 2 * decay, 2.25 + decay * decay])
        )  # Transfer function in s-domain

    w, S, phi = H.bode()  # Bode
    plt.figure(cnt + 1)
    plt.subplot(2, 1, 1)  #  Magnitude plot
    plt.title(f"Bode plot of transfer function ({level} decay)")
    plt.semilogx(w, S, label="|H(s)|")
    plt.legend()
    plt.grid()
    plt.subplot(2, 1, 2)  # Pase plot
    plt.semilogx(w, phi, label="<H(jw)")
    plt.legend()
    plt.grid()
```

```
    # Impulse response
    t, x = sp.impulse(H, None, np.linspace(0, 50, 500))
    plt.figure(cnt + 2)
    plt.plot(t, x)
    plt.title(f"Time response of spring (decay = {decay})")
    plt.xlabel("t")
    plt.ylabel("x(t)")
    plt.grid()
    pass


# 1, 2: Plotting impulse response for two different decay values
plot_impulse(0.5, "High", 0)
plot_impulse(0.05, "Low", 2)
```

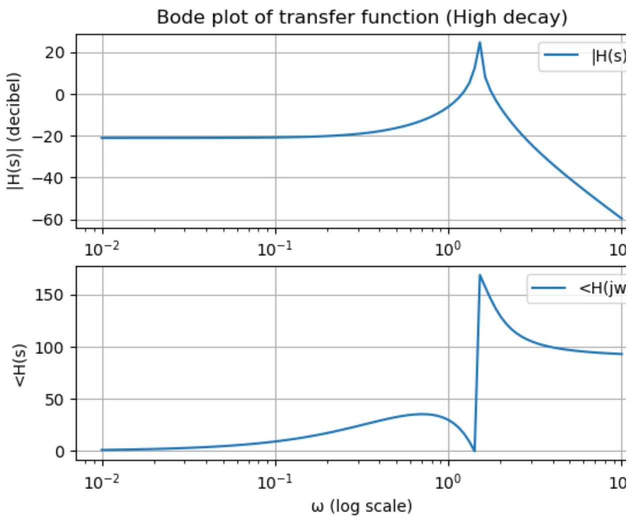

Figure 1: Bode Plot for decay 0.5

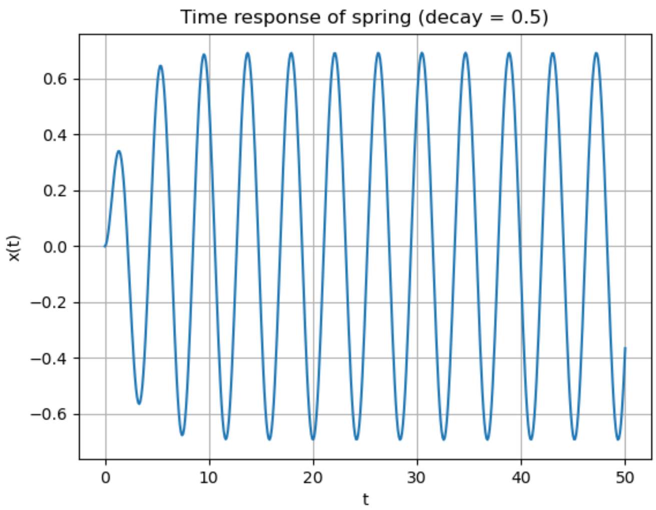

Figure 2: Impulse response for decay 0.5

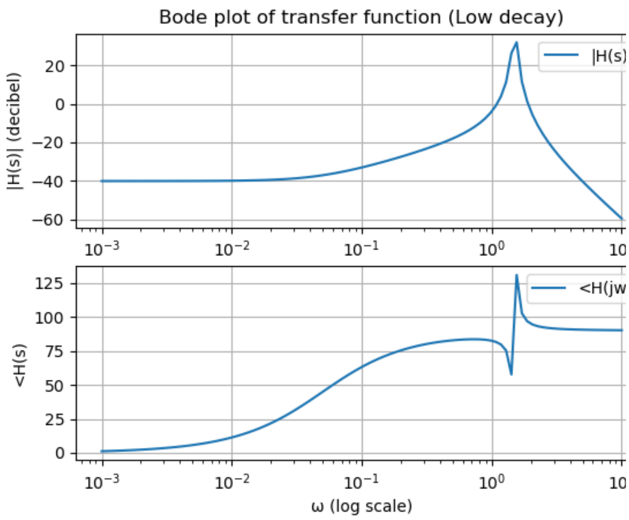

Figure 3: Bode Plot for decay 0.05

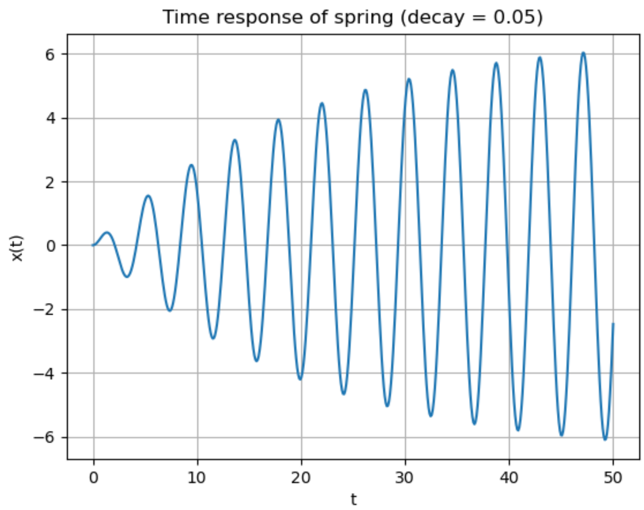

Figure 4: Impulse response for decay 0.05

2.1.2 Responses for various frequencies

Now we will vary the frequency of the input signal from 1.4 to 1.6 and analyze the output response of the system using scipy.lsim function

```
H = sp.lti([1], [1, 0, 2.25])  # Transfer function in s-domain
omega = 1.4
while omega <= 1.6:
    t = np.linspace(0, 50, 500)
    f_t = np.cos(omega * t) * np.exp(-0.05 * t)  # Different omega values
    t, x, svec = sp.lsim(H, f_t, t)

    # Output response
    plt.figure(5)
    plt.plot(t, x, label=f"w = {omega}")
    plt.title("x(t) for in frequency range", fontsize=12)
    plt.xlabel("t", fontsize=10)
    plt.ylabel("x(t)", fontsize=10)
    plt.legend()
    plt.grid()
    plt.savefig("5.png")

    omega += 0.05
```

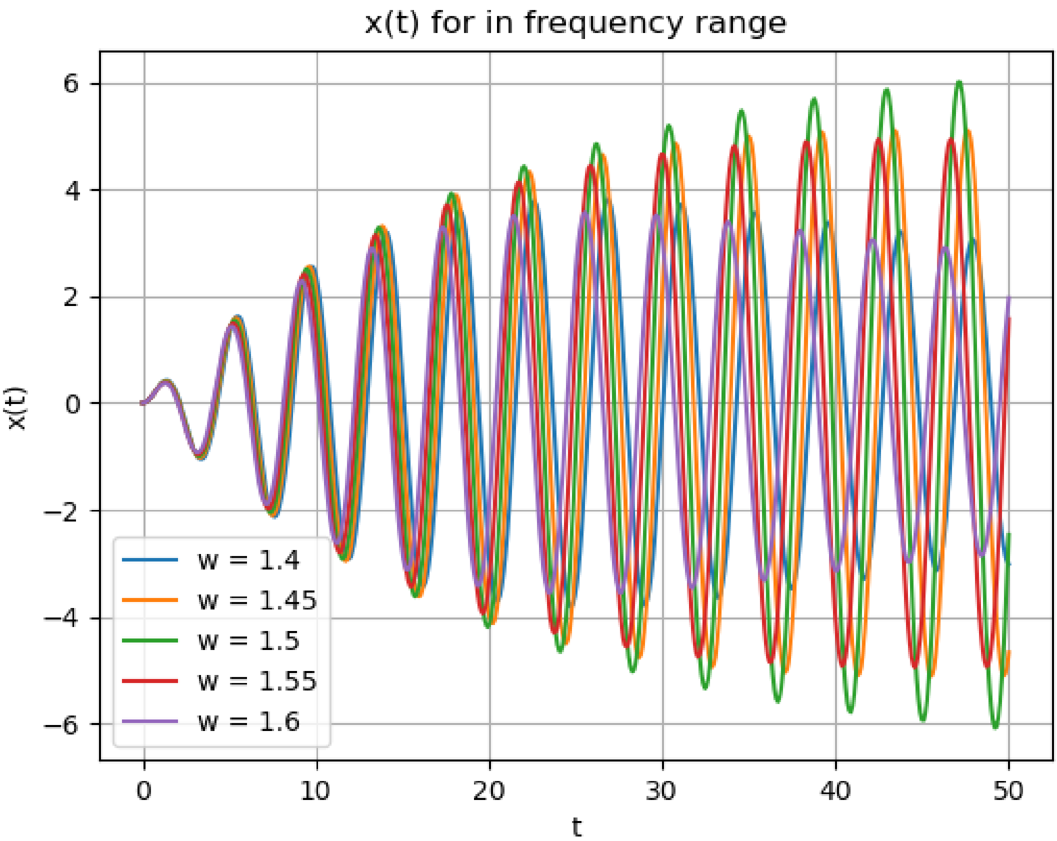

Figure 5: Output for different frequencies

2.2 Coupled spring problem

In this problem we have two differential equations and two variables to solve for. The equations are

$$\frac{\mathrm{d}^2x}{\mathrm{d}t^2} + (x - y) = 0 \tag{4}$$

$$\frac{\mathrm{d}^2 y}{\mathrm{d}t^2} + 2(y - x) = 0 \tag{5}$$

With initial condition as x(0) = 1 On solving both equations we get,

$$X(s) = \frac{s^2 + 2}{s^3 + 3s} \tag{6}$$

$$Y(s) = \frac{2}{s^3 + 3s} \tag{7}$$

```
# 4: Coupled spring problem
X = sp.lti([1, 0, 2], [1, 0, 3, 0])  # Transfer fucntions in s-domain
Y = sp.lti([2], [1, 0, 3, 0])

# Calculating x(t) and y(t) solutions for spring problem
plt.figure(6)
t, x = sp.impulse(X, None, np.linspace(0, 20, 500))
plt.plot(t, x, label="x(t)")
t, y = sp.impulse(Y, None, np.linspace(0, 20, 500))
plt.plot(t, y, label="y(t)")
plt.title(f"Coupled spring solution", fontsize=12)
plt.xlabel("t", fontsize=10)
plt.ylabel("function", fontsize=10)
plt.legend(loc="upper right")
plt.grid()
plt.savefig("6.png")
```

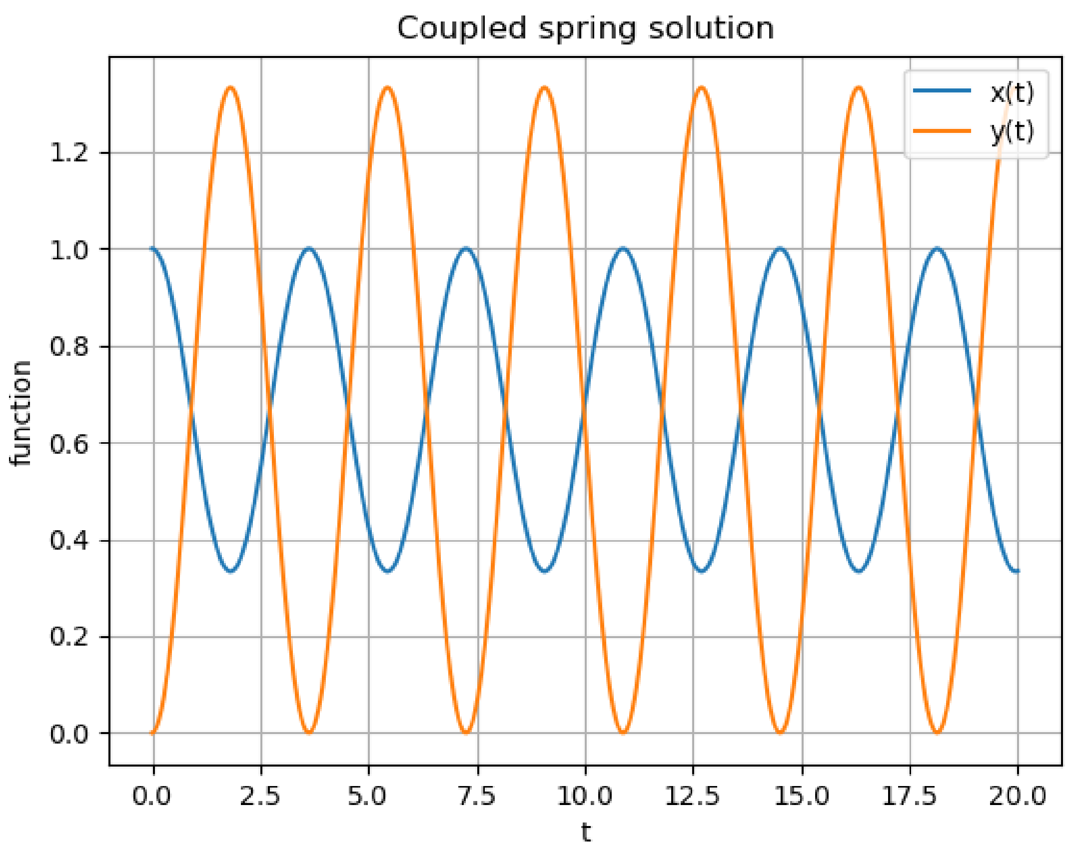

Figure 6: Solutions of coupled spring equations

2.3 Two port RLC Circuit

2.3.1 Bode plot of Steady State Transfer function

We now consider the case of an RLC Filter with the transfer function as shown.

$$H(s) = \frac{1}{10^{-12}s^2 + 10^{-4}s + 1} \tag{8}$$

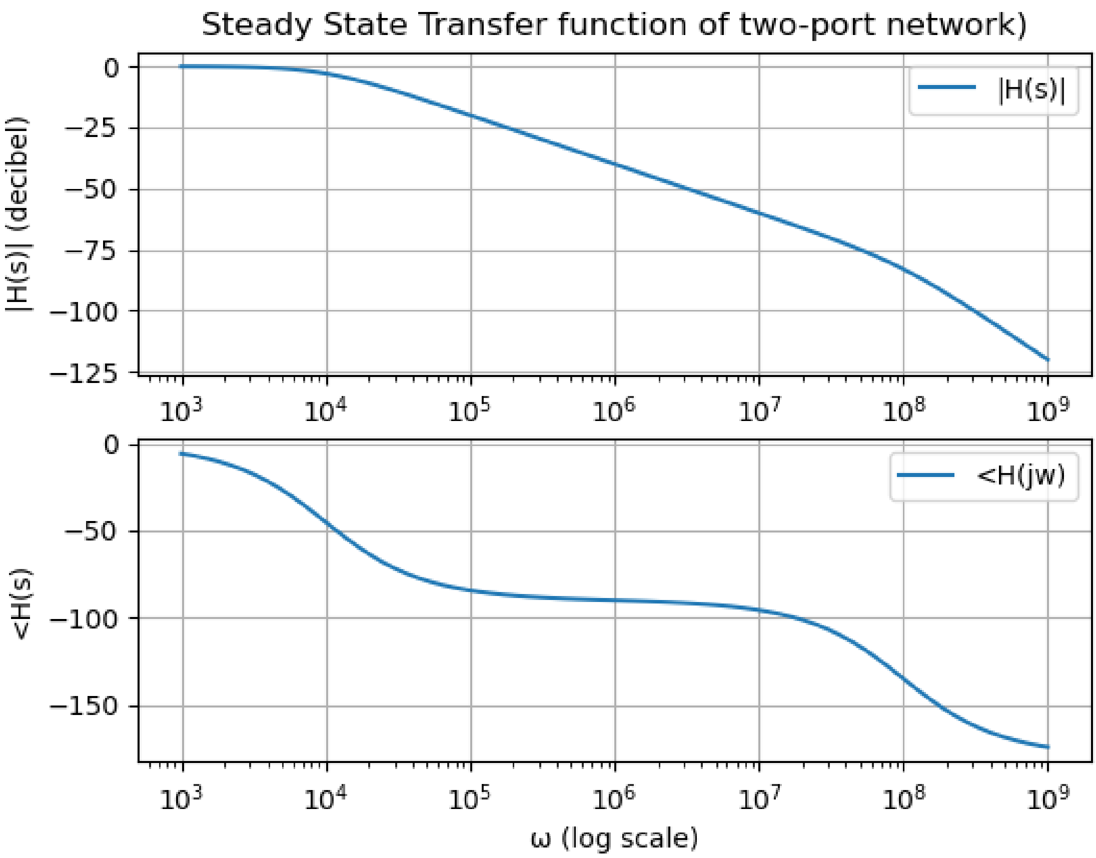

Figure 7: Bode plot of transfer function in RLC circuit

2.3.2 Output voltage for given input signal

The input is of the form

$$x(t) = \cos(10^3 t) - \cos(10^6 t) \tag{9}$$

```
t = np.arange(0, 20e-3, 1e-7)
v_t = np.cos(1e3 * t) - np.cos(1e6 * t)
t, x, svec = sp.lsim(H, v_t, t)

# Long time scale plot of Output
plt.figure(8)
plt.plot(t, x)
plt.title("Output voltage of RLC two-port network (long time)", fontsize=12)
plt.xlabel("t", fontsize=10)
plt.ylabel("Vo(t)", fontsize=10)
plt.grid()
plt.savefig("8.png")

# Short time scale plot of Output
plt.figure(9)
```

```
plt.plot(t[0:500], x[:500])
plt.title("Output voltage of RLC two-port network (short time)", fontsize=12)
plt.xlabel("t", fontsize=10)
plt.ylabel("Vo(t)", fontsize=10)
plt.grid()
plt.savefig("9.png")
```

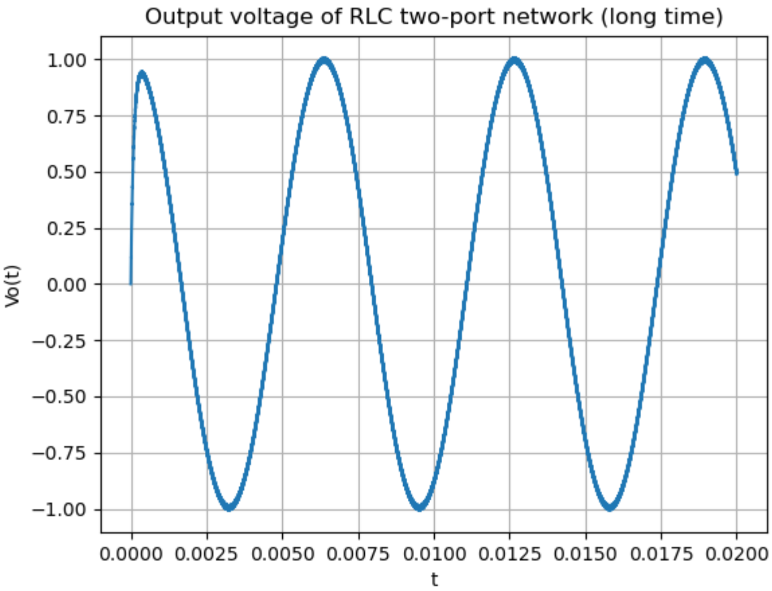

Figure 8: Output voltage for t<20msec

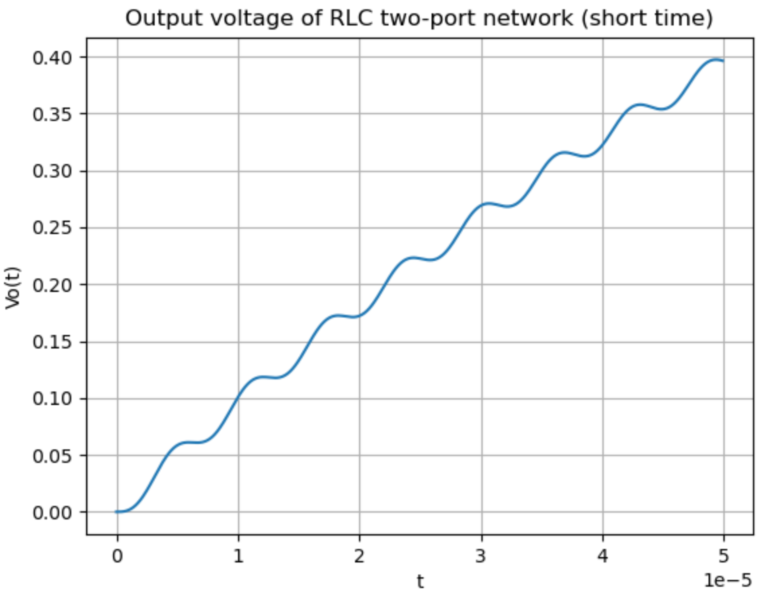

Figure 9: Output voltage for t<10µsec

3 Conclusion

The ripple in the signal in small time interval plot is because of the high frequency component. But the ripples are not visible in large time interval because the presence of low frequency component of which amplitude remained almost unchanged and dominated high frequency signal.